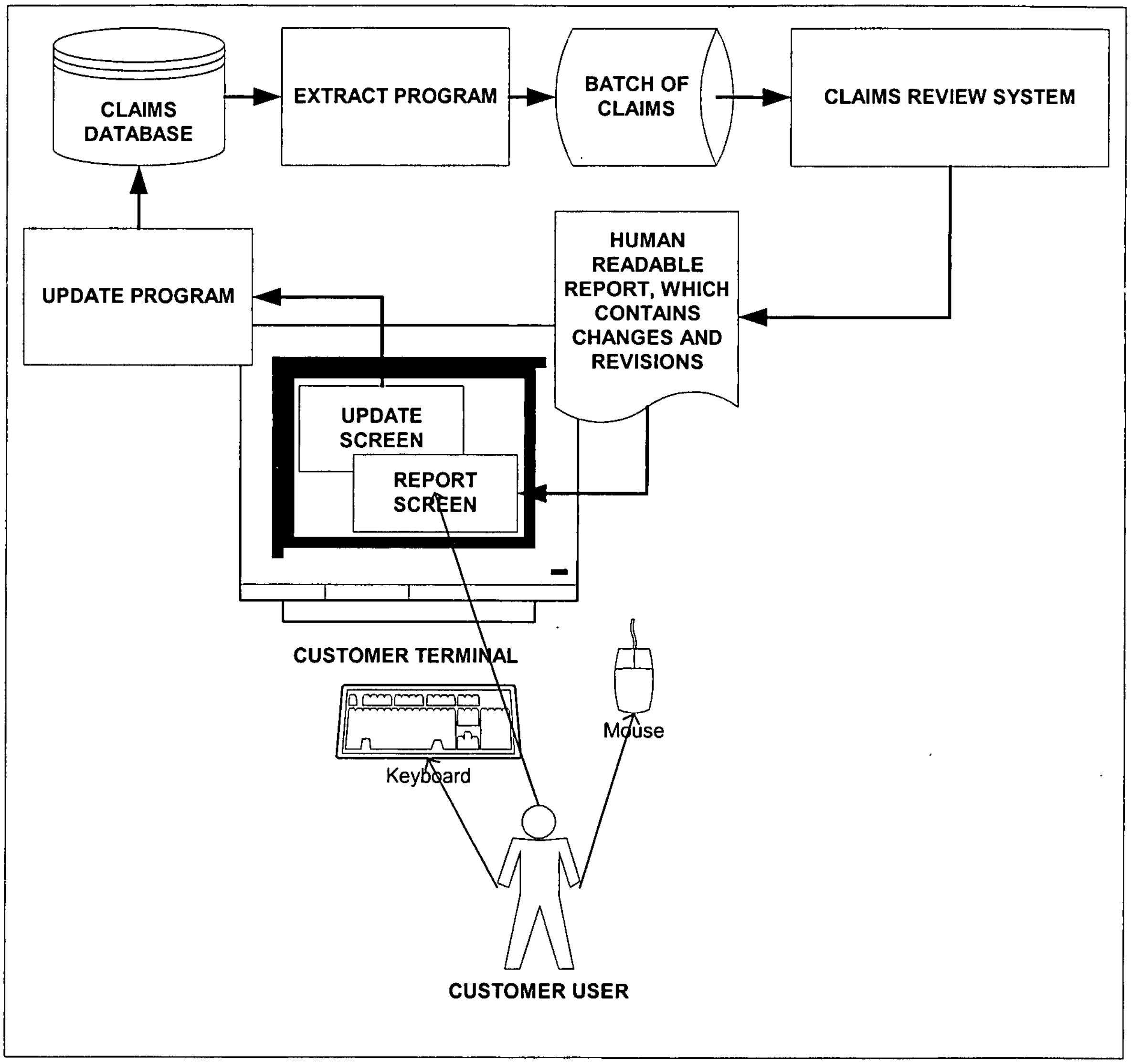
Fig. 1. Prior Art Manual Application



5

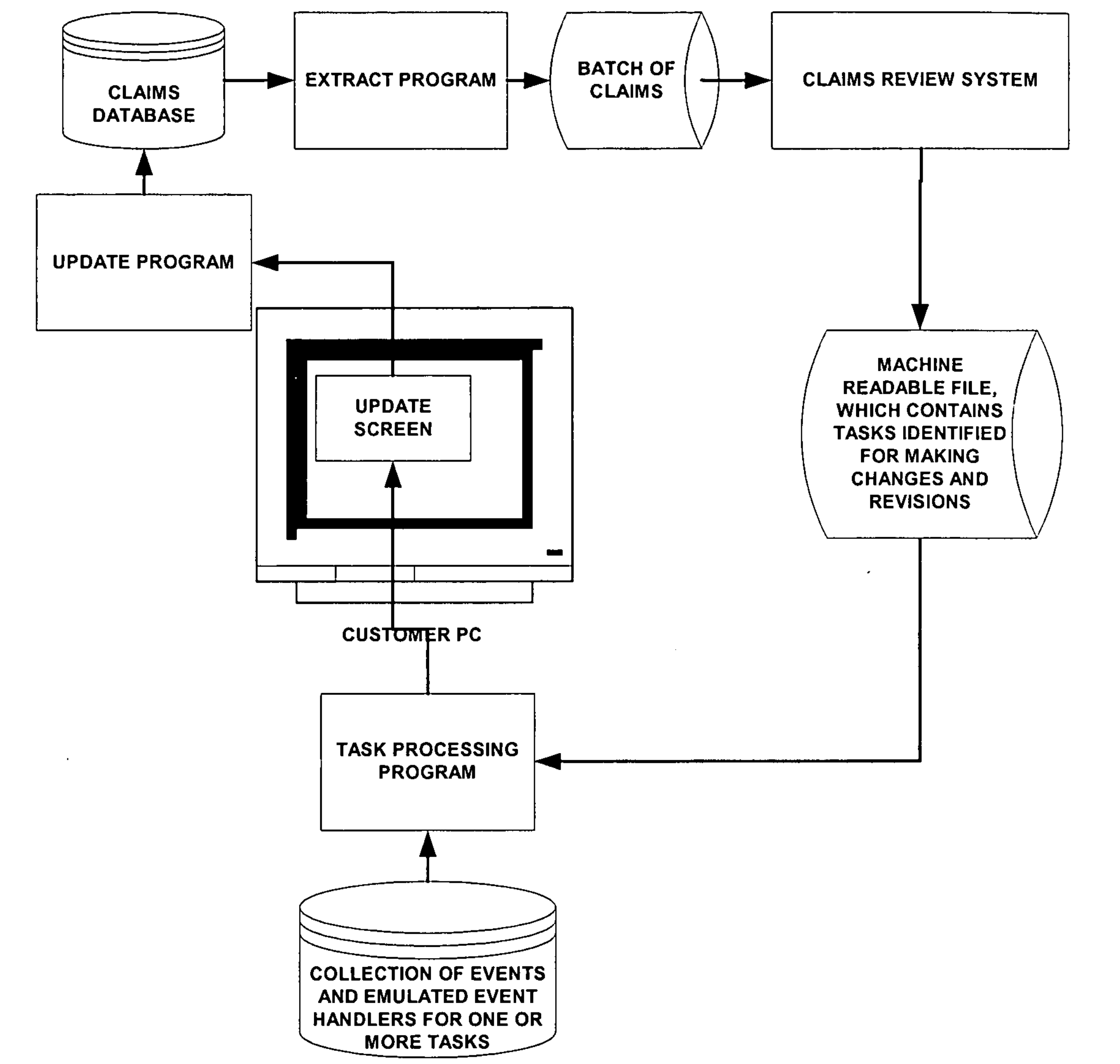
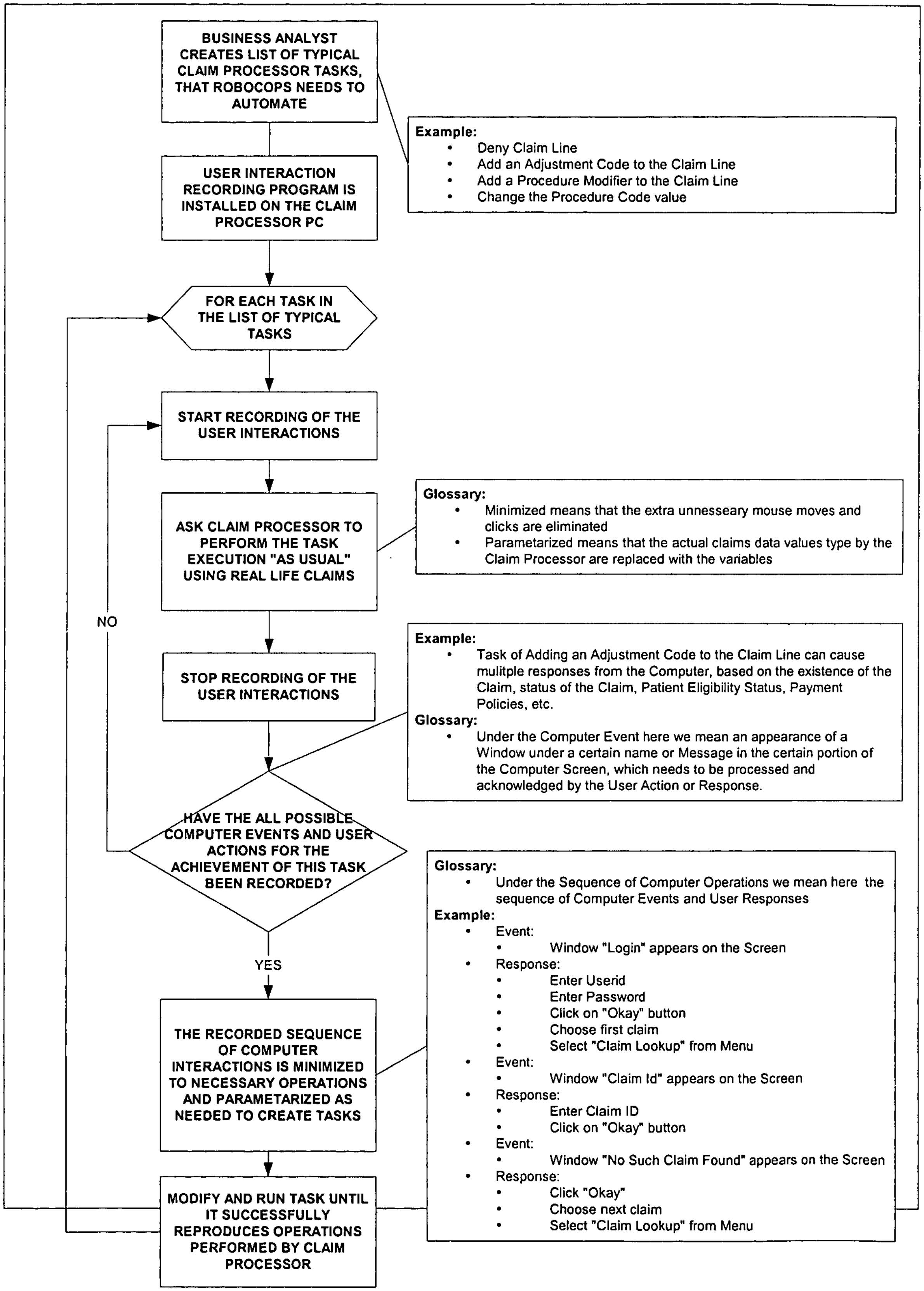

CLAIMS DATABASE
EXTRACT PROGRAM
BATCH OF CLAIMS
CLAIMS REVIEW SYSTEM
UPDATE PROGRAM
UPDATE SCREEN
CUSTOMER PC
MACHINE READABLE FILE, WHICH CONTAINS TASKS IDENTIFIED FOR MAKING CHANGES AND REVISIONS
TASK PROCESSING PROGRAM
COLLECTION OF EVENTS AND EMULATED EVENT HANDLERS FOR ONE OR MORE TASKS

Fig. 2 Process of Present Invention


BUSINESS ANALYST CREATES LIST OF TYPICAL CLAIM PROCESSOR TASKS, THAT ROBOCOPS NEEDS TO AUTOMATE
USER INTERACTION RECORDING PROGRAM IS INSTALLED ON THE CLAIM PROCESSOR PC
FOR EACH TASK IN THE LIST OF TYPICAL TASKS
START RECORDING OF THE USER INTERACTIONS
ASK CLAIM PROCESSOR TO PERFORM THE TASK EXECUTION "AS USUAL" USING REAL LIFE CLAIMS
NO
STOP RECORDING OF THE USER INTERACTIONS
HAVE THE ALL POSSIBLE COMPUTER EVENTS AND USER ACTIONS FOR THE ACHIEVEMENT OF THIS TASK BEEN RECORDED?
YES
THE RECORDED SEQUENCE OF COMPUTER INTERACTIONS IS MINIMIZED TO NECESSARY OPERATIONS AND PARAMETARIZED AS NEEDED TO CREATE TASKS
MODIFY AND RUN TASK UNTIL IT SUCCESSFULLY REPRODUCES OPERATIONS PERFORMED BY CLAIM PROCESSOR

Example:
- Deny Claim Line
- Add an Adjustment Code to the Claim Line
- Add a Procedure Modifier to the Claim Line
- Change the Procedure Code value

Glossary:
- Minimized means that the extra unnesseary mouse moves and clicks are eliminated
- Parametarized means that the actual claims data values type by the Claim Processor are replaced with the variables

Example:
- Task of Adding an Adjustment Code to the Claim Line can cause mulitple responses from the Computer, based on the existence of the Claim, status of the Claim, Patient Eligibility Status, Payment Policies, etc.

Glossary:
- Under the Computer Event here we mean an appearance of a Window under a certain name or Message in the certain portion of the Computer Screen, which needs to be processed and acknowledged by the User Action or Response.

Glossary:
- Under the Sequence of Computer Operations we mean here the sequence of Computer Events and User Responses

Example:
- Event:
  - Window "Login" appears on the Screen
- Response:
  - Enter Userid
  - Enter Password
  - Click on "Okay" button
  - Choose first claim
  - Select "Claim Lookup" from Menu
- Event:
  - Window "Claim Id" appears on the Screen
- Response:
  - Enter Claim ID
  - Click on "Okay" button
- Event:
  - Window "No Such Claim Found" appears on the Screen
- Response:
  - Click "Okay"
  - Choose next claim
  - Select "Claim Lookup" from Menu

Fig 3. Process Flowchart for Emulation & Recording of Event Handlers for each event of a task

5

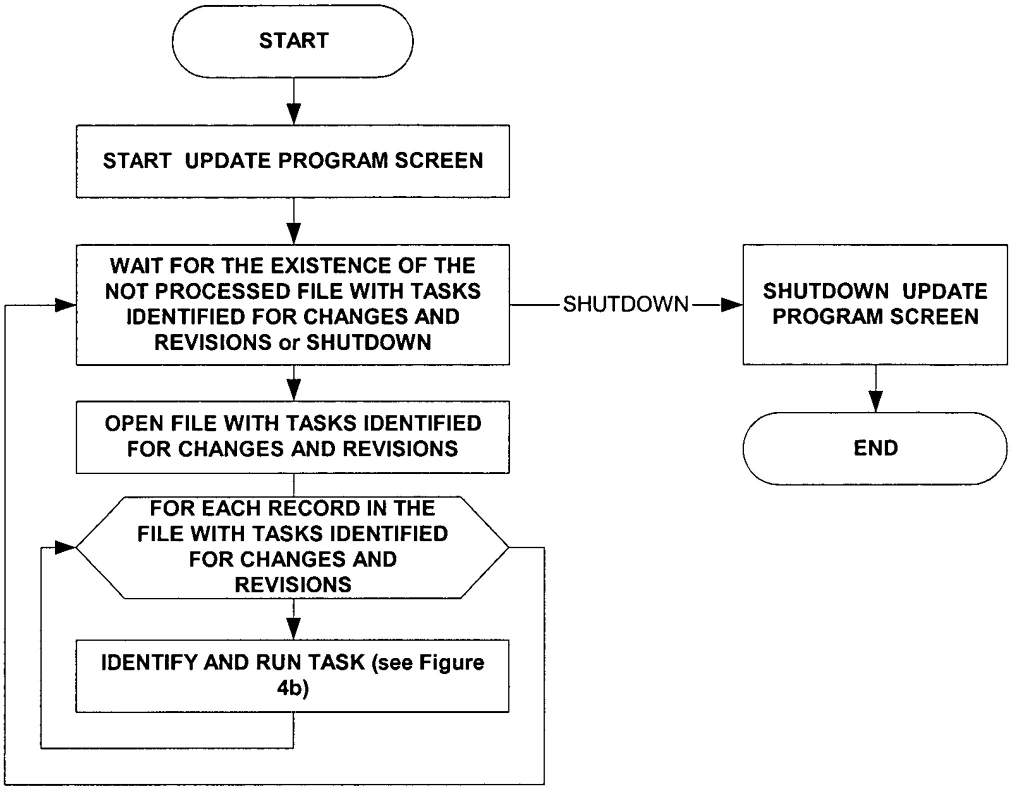

START
START UPDATE PROGRAM SCREEN
WAIT FOR THE EXISTENCE OF THE NOT PROCESSED FILE WITH TASKS IDENTIFIED FOR CHANGES AND REVISIONS or SHUTDOWN
SHUTDOWN
SHUTDOWN UPDATE PROGRAM SCREEN
END
OPEN FILE WITH TASKS IDENTIFIED FOR CHANGES AND REVISIONS
FOR EACH RECORD IN THE FILE WITH TASKS IDENTIFIED FOR CHANGES AND REVISIONS
IDENTIFY AND RUN TASK (see Figure 4b)

Fig. 4a Flowchart for Task Processing

5

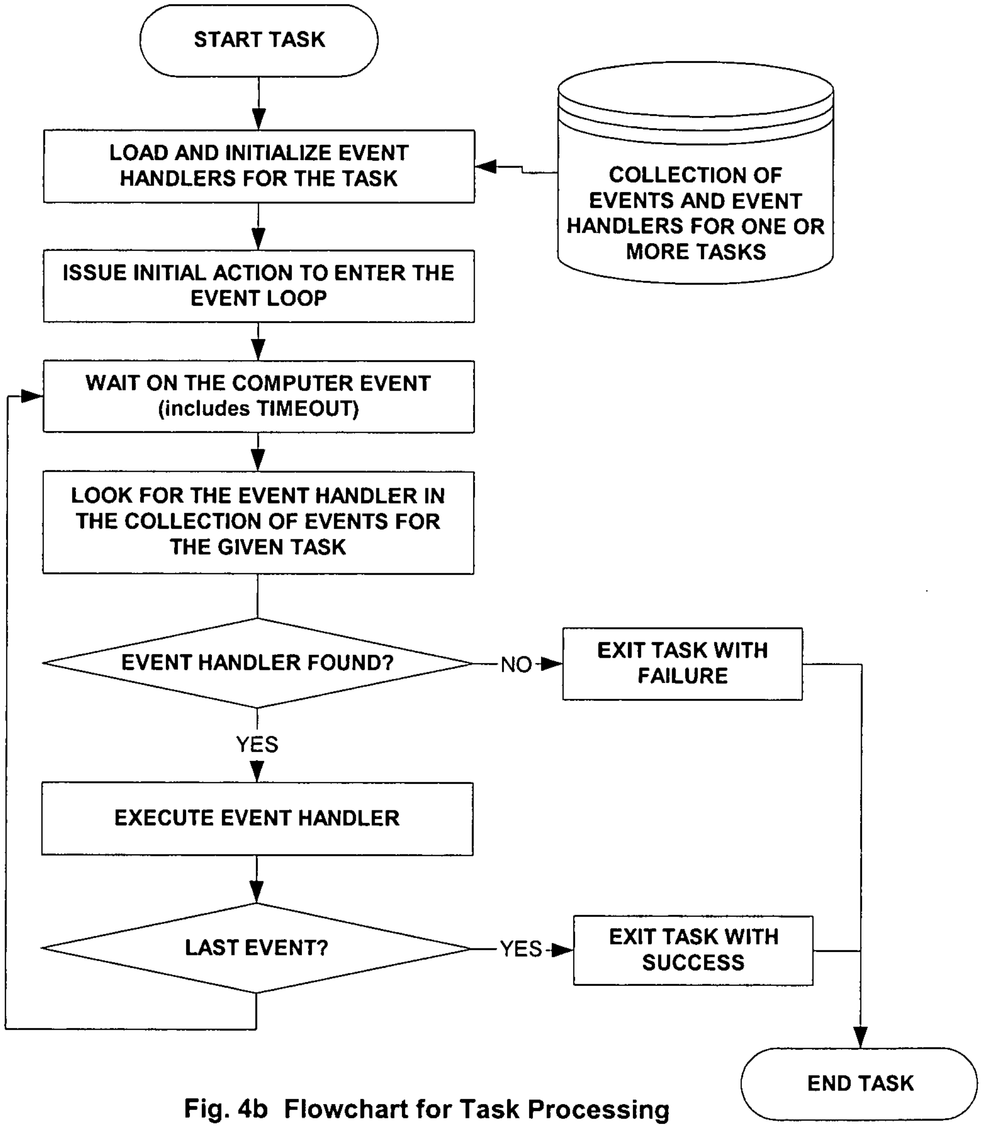

START TASK
LOAD AND INITIALIZE EVENT HANDLERS FOR THE TASK
COLLECTION OF EVENTS AND EVENT HANDLERS FOR ONE OR MORE TASKS
ISSUE INITIAL ACTION TO ENTER THE EVENT LOOP
WAIT ON THE COMPUTER EVENT (includes TIMEOUT)
LOOK FOR THE EVENT HANDLER IN THE COLLECTION OF EVENTS FOR THE GIVEN TASK
EVENT HANDLER FOUND?
NO
EXIT TASK WITH FAILURE
YES
EXECUTE EVENT HANDLER
LAST EVENT?
YES
EXIT TASK WITH SUCCESS
END TASK

Fig. 4b Flowchart for Task Processing

Figure 5. Example of a Typical Task

The following simplified example illustrates the collection of Event Handlers which are required for the execution of the task "Add Adjustment Code to the Claim Line".

Event	**Handler**
<initial>	Select "Claim Lookup" from the Menu
Claim Lookup Window appears	Enter Claim Id Click on "Lookup" button
Claim Display Window appears	Click on the "Lines" button
Claim Lines Display Window appears	Go through all the lines, matching the line number against the given. If the line number found, emulate the double-click on the line. Set "Exit Status" to "Failure – No Line". If the line number not found, click on the "Close" button.
Claim Line Edit Window appears	Enter new Adjustment Code into the adjustment code field. Set "Exit_Status" to "Success". Click on the "Close" button.
Claim is Pended Window appears	Set "Exit Status" to "Failure - Pended". Click on "Okay" button.
Claim Not Found Window appears	Set "Exit Status" to "Failure – Not Found". Click on "Okay" button.
<timeout>	Set "Exit Status" to "Failure – Timeout". Select "Exit" from the Menu

Figure 6a. Example of typical flow of events

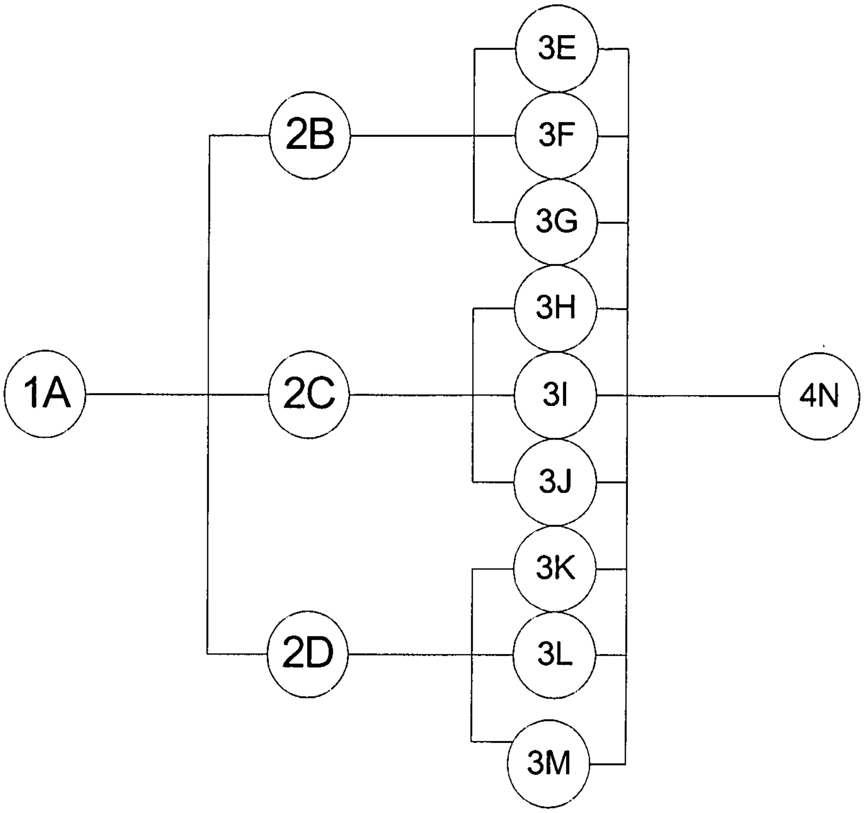


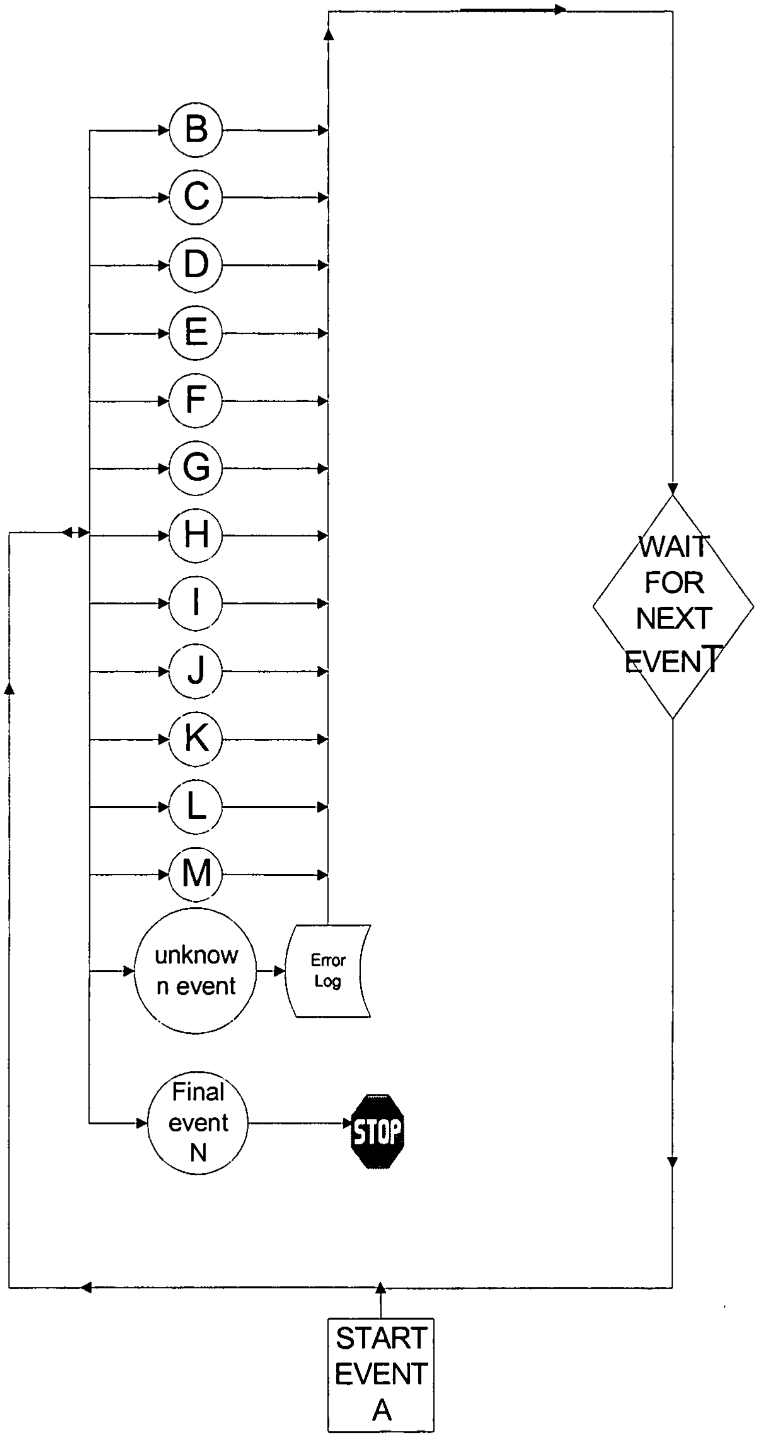
Figure 6b. Prior Patent:

1. A --> B --> E --> N
2. A --> B --> F --> N
3. A --> B --> G --> N
4. A --> C --> H --> N
5. A --> C --> I -->N
6. A --> C --> J --> N
7. A --> D --> K --> N
8. A --> D --> L --> N
9. A --> D --> M --> N

With only 3 steps, and three events potentially occurring at each step Rigid Sequence Programming would have to account for nine potential sequences;

If A --> C --> M occurred, the program would fail.

FIGURE 6C: PATENT PROCESS



The invention requires the programming of events and event handlers. The event handlers wait for events to occur, respond, then waits again. If an unknown event occurs, the program can recognize and handle it by noting the event and writing it to a report, without disrupting the continuation of the program. The programming required by this type of programming is minimal compared to the sequences required of the prior patent.

5